Exhibit 36

trials@uspto.gov

571-272-7822

IPR2014-01462, Paper No. 60
IPR2014-01469, Paper No. 55
January 5, 2016

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

EMC CORPORATION,
Petitioner,

v.

ACQIS, LLC,
Patent Owner.

Case IPR2014-01462, Patent 8,041,873 B2
Case IPR2014-01469, Patent RE42,814 E

Held: December 8, 2015

BEFORE: MICHAEL P. TIERNEY, MICHAEL J. FITZPATRICK, and ROBERT J. WEINSCHENK, Administrative Patent Judges.

The above-entitled matter came on for hearing on Tuesday, December 8, 2015, commencing at 1:00 p.m., at the U.S. Patent and Trademark Office, 600 Dulany Street, Alexandria, Virginia.

bits of PCI bus transaction. We agree that there has to be a PCI bus transaction somewhere, but the only thing that needs to flow across that LVDS channel are address and data bits. You don't have to have the complete transaction. You don't have to have the complete configuration file that a PCI bus transaction might have. You don't need the command information or the byte enables. That's not what they chose to claim. If they had wanted to claim that, they could have said conveying the PCI bus transaction in serial form. Or there's a bunch of different ways they could have claimed it.

If we can go to slide 7, just so we are clear, the language in claim 54 of the '873 is slightly different, we admit, but there again, it's actually even broader because what has to be transmitted there is an encoded serial bit stream of PCI bus transaction. So you just need a bit stream of the transaction. It doesn't even specify that it has to be address or data. We know that from looking at dependent claim 61 which is highlighted on this slide 7 as well that specifies that the encoded bit stream comprises encoded PCI address and data bits.

So claim 54 has got to be broader than claim 61 and covers a bit stream that's not even address or data bits. So that's really the third response to patent owners was we read their argument as requiring that you have to transport across the LVDS channel the entire transaction, and that's not simply what they claimed.

addressing and PCI's physical addressing. But the claims don't say physical addressing. They don't say you can't use virtual addressing. Again, it goes back to the claims which only says you have to transmit address bits of PCI bus transaction.

So let's look again at slide 30. So here is where we've put together some testimony from their expert in a quote from the patent owner's response which lead us to believe that they are arguing that not only do you have to have all of the address bits but you have to have other things from the PCI bus transaction for the claim to be met. In the top left-hand corner they suggest that you can't have a PCI bus transaction unless you send the configuration information. The next one down, Dr. Lundenstruth, their expert suggests that the claims require all address and data phase information including the PCI data and byte enables during the data phase. And there's several other quotes we've got here on the slide. Again, we don't believe that that's a requirement in these claims and we think that all the Board needs to find is if there's transmission of some bits of address and data, and we get that by looking simply at the claims.

Go to slide 31. Again, we've highlighted the relevant portions here on slide 31 of the claims which don't ever say you have to submit all of the address or data bits. Earlier I mentioned the claim differentiation point between claim 54 of the '873 patent and claim 61 of the '873 patent which suggests that certainly for the claim 54, at most, you have to have a bit stream and it doesn't

even have to be address and data. So again, we think that that argument undermines their theory that you have to have all address bits.

If there's no further questions about Horst, I wanted to quickly touch on a few points about Bogaerts.

JUDGE WEINSCHENK: That's fine.

MR. BUROKER: Could we go to slide 53. So the analysis for Bogaerts is very similar to what we just talked about but let me reorient you. Instead of having the TNet network, what you have in those large circles in the top left Figure 13 that's shown is the SCI network.

JUDGE WEINSCHENK: I actually have a question before we get into too many details here. Are you relying on Figure 13 embodiment or Figure 15 embodiment or are you arguing they should be combined? I'm not sure I follow the argument that you are making here because you kind of cite to both of them.

MR. BUROKER: Your Honor, Figure 13, we believe, shows all of the elements of the claim. Figure 15 is another embodiment where they have taken a Pentium chip and included various components so it's an alternative attached computer module and it teaches that it can be plugged into an SCI ring. And so Your Honor, I think it's a little bit of both. We are arguing Figure 13 discloses everything.

the standard. It's always a physical address. Never a virtual. That's why it matters. And we've heard some argument about the claims don't say virtual or physical. Yes, they do. PCI address is a physical address by definition.

So when in ServerNet or TNet or Horst, however you want to refer to it, does the physical address first exist? Keep in mind their expert wasn't shown this document. We weren't shown this document. We had to dig it out of a pile of litigation stuff that they turned over to us. But what we know is you start up front with the ServerNet address. You run it through an address validation and translation table and at the end you get the local physical address.

JUDGE WEINSCHENK: Why is that problematic? If you are sending address bits over the TNet that can -- maybe they are encoded in some way but on the other end you could decode them and get a PCI address, why is that not covered by the claims?

MR. STACY: It absolutely doesn't because you are never starting with an address here. So their expert describes it before he changed his position in his original declarations, he describes it as Scrabble cubes. I've got a PCI address, I have got it spelled out here, 32-bit word, and I take it, I put it on my serial line. So that's the whole point. I'm going from parallel, I'm putting on a serial line. And on the bottom end I have got to be

able to reassemble my word. I have got my rules. I know what to do.

On all of the TNets, all of the Bogaerts, they all have the same issue is that there is no physical address that's then scrambled and then put back together down here. You start with a made-up virtual address, manipulate it and then you have some kind of smart device, it's those buses in TNet, you see it in those figures on what goes on the bottom side by the PCI device that creates the PCI address for the first time.

JUDGE WEINSCHENK: I guess I'm struggling to understand though why that matters, right. Why do you have to have a physical address to start with? Why can't it be virtual or some other address? As long as it's information that you could on the other end translate into a PCI address, why does that not meet the claims? I think we agree that what you are transmitting across the channel doesn't actually have to be in perfect PCI format, right? You can change it.

MR. STACY: Format, I agree. I do not agree that the content can be changed. The content has to be the same. In other words, think of my Scrabble cubes. I have got a word here. I can jumble it, do whatever I want to in the middle. But on the back end I have got to be able to build that same word again. That's what the claims require. If you don't do that, you have destroyed your PCI standard. You have destroyed the very purpose of this invention. The whole point is you start PCI, whether you start

here or down here, it makes no difference, whether it's the CPU or the printer. You start with the PCI address that's in this parallel slow form, serialize it and then take it back to the PCI form at the other end.

JUDGE TIERNEY: I would like to go back to the language of the claims because I think we are reading a few things in and I want to make sure it's okay to do so. Let's start with the '873 patent claim 54. We are talking about encoded serial bit stream. Why do I want to read in the words physical address?

MR. STACY: It is an encoded serial bit stream of peripheral interconnect bus transaction. The Board has already said that the PCI is important, that it's part of it. So you have got to go to the standard. The PCI standard does not allow a virtual address. So you would be reading PCI standard out of claim 54. So in other words, if you put a virtual address in, it's no longer a capital PCI bus transaction. It's something completely different. And that's just straight out of the standard. Dr. Lindenstruth references that several times and we'll talk about that.

Now, there's something interesting here. They keep going back to this claim differentiation doctrine on claim 61 which showed up in the reply for the first time. You heard it the same time we did, but I would like to the address that.

JUDGE TIERNEY: Before we move on, you mentioned that basically you have to have all of the content

Here we have the PCI standard. And there are two in exactly two different addressing schemes. You see it every day when you buy a laptop, 32-bit processor or 64-bit. This was the days of the 32-bit when this was drafted. But the PCI standard is 32 or 64. You send 31, it breaks. You have to have the full piece.

JUDGE WEINSCHENK: So what if you sent 31, like we send a shortened number here and it knows at the other end to add in the 32nd bit? Doesn't that convey sufficient information for a PCI bus transaction?

MR. STACY: No, it absolutely doesn't. Because what you are doing is saying that the transaction happens at the end. The printer prints and that's true. The printer prints. My CPU says print this document. The printer prints. But the whole point of this invention and these claims is that they are centered around dealing with a standard. All of the computers out there have PCI. And so the point was, how do you speed up PCI? And the point is you take it from parallel to serial and then back to parallel.

And that's the point. If you somehow say, well, okay, we are just going to do away with PCI, that's not the invention and that's like saying I'm going to buy a new Windows machine and I'm not going to use PCI. I'm just going to use a 30-bit rather than 32. It doesn't work. You violated the standard. Just like internally, if you don't know to put 2 instead of 8, your four-digit number does you no good. You have to have the PCI standard.

And the Board has already gone to that in saying that PCI isn't part of the standard. By the way, when you looked at their expert and we asked their expert, what do you think about the Board's construction? He said it's broad, it's reasonable and it's not inaccurate. That's probably the best you'll ever get from a professional expert. I didn't press him to say will you admit it's accurate but he just kept saying it wasn't inaccurate. He wasn't going to quarrel with the Board. So we've already answered that question. It must be by the PCI standard. Anything less is not the PCI standard.

And an important thing here about 61, they said 61 somehow offers claim differentiation. This is something their expert came up with. Well, if you look at the standard, there are three types of information included in every PCI transaction. Three types. There's an address, there's data and then there's control. That's straight out of the standard. You have those three things.

There is a special type of PCI transaction known as interrupt acknowledge. In an interrupt acknowledge, the address is zeroed out. It's null. So if you look at claim 54, it says all PCI bus transactions. That covers everything. What does claim 61 do? Claim 61 narrows it and says this PCI transaction has to have encoded PCI address and data bits. Claim 61 actually carves out those interrupt acknowledges. It carves out a specific type.

And when you look at 24, 24 and 31 are at the narrower group of PCI transaction types. They are talking about the transaction types that have an address and those are non-interrupt-based ones.

JUDGE FITZPATRICK: If I understand this correctly, then, it's your position that a PCI bus transaction, that term in claim 54 does not require address bits and claim 61 expressly does and that is why claim 61 is narrower and differentiated?

MR. STACY: Correct. Practically it carves out the interrupt acknowledge PCI transactions in claim 61.

JUDGE FITZPATRICK: Is there any -- there's also a rule about claim construction that terms are to be given some meaning and not read to be extraneous. Is there any reason for the last three words in claim 61, "and data bits"?

MR. STACY: No. I mean, with claim differentiation, there's also significant warning that what the claim differentiation tail doesn't wag the construction dog. When we pressed their expert on the basis for that construction, I asked him about the spec, I asked him about the file history, and he had nothing else to support the construction. When you look --

JUDGE FITZPATRICK: You had also mentioned that it's in the reply. So maybe there is no evidence but I'm going to ask anyway, is there evidence from your side about this -- what was the term you called? Interrupted?

MR. STACY: It's an interrupt acknowledge.

JUDGE FITZPATRICK: Is there evidence in the record that that refers to a PCI transaction that lacks address data, address bits?

MR. STACY: So your question was is there anything in the record, and I'm in a quandary there because I didn't get to put anything in. So in the record, no, because the first time anybody saw this was in the reply. But with your permission I can show you something that's in the record but not specifically briefed. This is Exhibit 2001 on page 37.

JUDGE FITZPATRICK: Exhibit what?

MR. STACY: 2001. I'll introduce what the document is. It is the PCI local bus standard Chapter 3 and it's titled Bus Operation. And --

JUDGE FITZPATRICK: I'm there with you.

MR. STACY: The piece we are looking at is Section 3.1.1, Command Definition. And right up top you see the initial command, the interrupt acknowledge. So the interrupt acknowledge is a big part of the PCI because you have got to get the processor's attention or you've got to get the device's attention. Then down at the bottom we highlighted for you, the address bits are logical don't cares during the address phase. So keep in mind the PCI standard has two timing requirements and they have what they call a data phase which the data is transmitted, so it's a time, and then the address phase. And in the address phase they send the address and the control bits. So

during that phase the control bit, if it's an interrupt acknowledge, you can see that the address bits are don't cares. So that's the way claim 61 is narrowed. Does that answer your question?

JUDGE FITZPATRICK: Thank you, yes.

MR. STACY: So one thing I would like to make sure that as we go through this discussion that you keep in mind 54 is different than 24. Fifty-four requires the encoded serial bit stream of peripheral component Internet connect bus transaction. That includes the information necessary to make a PCI transaction under the defined standard. That includes data, that includes address except for the interrupt acknowledge, and that includes the control bits every time. So you didn't hear a word about control bits, period. There is no such thing -- just put it bluntly, there is no such thing as a PCI transaction that does not have control bits. To carve control bits out of claim 54 and 61 is to make sure it does not comply with either the standard of a PCI or the purpose of the invention.

So when we talk about 54, you have to talk about control bits. And not once do they talk about Horst or Bogaerts transmitting any type of control bits. Just doesn't happen. And it doesn't happen in those systems because they are actually radically different systems. We went and got our expert because he was an expert in the prior art that was cited. Those systems are for massively parallel computer systems. Bogaerts, for

example, was built to process the data at the CERN supercollider because nothing else existed.

And our expert, Dr. Lindenstruth, was the guy on the ground building that system. So he knows what these systems do. And he said over and over you can't take these virtual addresses on these serial networks and just put PCI on them. It will destroy it. It will actually cause address collisions. It will render them inoperable. The reason people like Horst and TNet came up with these systems is because PCI did not work. They had to come up with a proprietary bus. And that's said on the first page of Horst. Horst next to the picture of PCI says we considered all of the standard buses. We looked at all of them and decided all of them were insufficient. We designed our own. And Bogaerts is the same way. When you look at them, SCI and the TNet systems are both massively parallel systems. They have a proprietary network in the center that actually does the serial transmissions.

JUDGE WEINSCHENK: Your patents also changed the bus too, right? You said a parallel bus doesn't work so we are going to change it to a serial bus. So it's also not a standard PCI bus.

MR. STACY: You have got a standard PCI bus, you have got the invention, depending on how you do it, but you have got the new bus in the center, and what that's doing is taking the parallel 32 bits coming down, putting them on a serial bus,

moving them and then taking them back out. It's an hourglass when you think about it, here, here, transmit, go back out.

JUDGE WEINSCHENK: Explain to me, I think I want to make sure I understand exactly what Horst does, then. If you have got a CPU that wants to send data to a PCI device, it has to send the data and it has to send an address to where that data is going to be stored in the PCI device?

MR. STACY: Correct.

JUDGE WEINSCHENK: How does it know, if it doesn't have the address of the PCI device originally, how does it know where to send it to?

MR. STACY: It doesn't know that it's a PCI device. It just knows it's sending to that printer right over there. You have got whatever protocol is being done up top, in Horst it's TNet. You've got a TNet protocol. You may have something proprietary coming off the processor. There's no PCI transaction that exists on the processor. It's then -- in Horst it's then translated using these virtual address tables sent across the serial bus, picked up by another smart device down here that generates the physical address and ships it off.

But I understand the appeal of saying, well, I say print and it prints, therefore, it must be done. And if our claim was that broad, that might be a problem, but the claim is actually to solve a specific problem with systems using PCI. And that's the reason that PCI standard matters.

communication protocols also. But the point here is -- actually, this point is there's no intervening PCI bus. But when you look at this and when you read it, there is a PCI transaction here that is Scrabbleized and then put on the serial link and then recreated back here. There's nothing inconsistent with the claim here.

And again, it looks -- you can abstract any computer system to this level. You've got to look under the covers. If you are going to comply with the PCI standard, which I believe Figure 8 is a PCI limited embodiment, I would have to confirm that, you need to take the PCI transaction on the front end, serialize it and do the PCI transaction on the back end. That's this embodiment. By the way, if you look at this, this is a computer system. Not this massive parallel system.

JUDGE WEINSCHENK: Does there need to be a PCI bus on the CPU side or does it just have to be something you can call a PCI address and data or do we actually need a bus there? I think you have argued in your papers that you need a bus. That's what I'm curious about because this one doesn't show a bus on the CPU side.

JUDGE TIERNEY: Let me direct you to page 5 of your patent owner response.

MR. STACY: I think it may depend claim by claim.

JUDGE TIERNEY: Patent owner response of the 1462 case.

MR. STACY: Page 5 you said?

JUDGE TIERNEY: Yes. Figure 8 illustrates that.

MR. STACY: Correct, there's no PCI bus.

JUDGE TIERNEY: And consequently the TNet link does not generate PCI bus transactions or PCI transactions at the CPU side.

MR. STACY: Correct. I think that's a description of Horst. You don't have to have the PCI bus. The PCI bus is a good indicator. But what we know from Horst, their own expert agrees, there is no indication there's a PCI transaction on the processor side within Horst.

JUDGE TIERNEY: So can you relate that information that we just discussed that is on the CPU side there's no PCI bus with Figure 8 of your patent in the '873 patent?

MR. STACY: So on Figure 8 of the '873 patent, you have to look at the text. It's talking about a PCI transaction. And so the claims don't require a bus. They require a PCI transaction to take place. You are looking at the standard the whole time. I'm putting a transaction together that looks like a PCI transaction. So all of my PCS-compliant drivers know how to read it. So I have created my PCI transaction. That's why it matters, my drivers can read it. And then I'm taking it at a parallel form, putting it into serial form and then back to parallel.

The point about Horst is that Horst doesn't have a PCI transaction that is then serialized. And one indicator is that there's no PCI bus but it makes no difference whether there is.

You could generate a PCI transaction without a bus. The point is and when you looked at the -- remember the sheet that I showed you, the transaction is defined on what data goes where.

JUDGE TIERNEY: But the way it was worded was basically what I was reading was because there's no PCI bus on the CPU side, the TNet link does not generate PCI bus transactions.

MR. STACY: That is the description of Horst there.

JUDGE TIERNEY: Then I look over at Figure 8 and I don't see a PCI bus on the CPU side.

MR. STACY: So in Figure 8 the illustration -- I mean, all of these are kind of at cartoon level. Figure 8, when you match it with the description, it's talking about the type of transaction. So you are starting with your PCI transaction and converting it. There are a lot of things that aren't shown in a seven-box drawing. But the point is if you are trying to match it to the claims, you are taking a PCI transaction and serializing it. That's what you are starting with.

JUDGE WEINSCHENK: Can you explain to me again how we know in Horst that the CPU doesn't start with a PCI transaction and scramble it into a TNet and then re-unscramble it to a PCI? So the lack of a PCI bus is not determinative is what I'm hearing you say.

MR. STACY: That's correct. The lack of a PCI bus is indicative but not determinative. In this particular instance they

And that's not really entitled to much weight. He's not an expert in these types of systems. And again, that came in the reply brief when we didn't really have a chance to address it. We went through that about putting in more evidence to address it.

But the point being here there's absolutely nothing in TNet or Bogaerts that they argue shows that you are putting a PCI address on that serial link. That's the reason they walk away from that and go to obviousness and use that language, it's simple, it's trivial. Any time anybody starts using those big words, you know that they are in trouble. But then they walk away from that and say, well, if you had two bits of a PCI transaction, that would be enough. Well, even if that's right, it doesn't address claim 54 because claim 54 requires a PCI transaction. Got to have control bits.

And for claim 24 and 31, the address, but you got to have them all because you are going by the PCI standard. Go back to your phone, because you had a standard, you know what it means. The PCI standard is the same way. PCI says 32 bits. These guys for these parallel systems don't want to use physical 32-bit addresses like the standard describes. Why? You get collisions.

And if you look at the front of Horst, the front of Horst, I think, tells you everything you need to know, is that he considered PCI and said no. I'm going to build my proprietary link in the center because they weren't interested in speeding up a

PCI standard transaction. They were interested in hooking up massive systems.

JUDGE FITZPATRICK: I have a question about we can look at claim 54, and the phrase "encoded serial bit stream of peripheral component interconnect PCI bus transaction," in that phrase would it be fair to summarize your construction of the word "of" as "from" and petitioner's as "for" such that you are arguing, I think, that the encoded serial bit stream must have originated from a PCI to begin with. Whereas, petitioner is arguing that in Horst you meet that limitation because that bit stream encodes for a PCI bus transaction.

MR. STACY: I would quarrel with any indication that we said it has to originate. I'm not sure what originate means. In this instance, what that claim requires is that it is a PCI bus transaction. That is defined by the standard. You know the three portions: Address, data and control. And you have to take that, whatever that would be in a parallel form and you are going to encode it, encode a serial bit stream. So I'm taking the transaction data, I don't think it matters where it originates from, then I'm going to take it, I'm going to serialize it and then send it out over my serial bus for speed purposes. That's why you're doing all this.

JUDGE FITZPATRICK: Let me rephrase. I didn't mean to put any special meaning to originate. Let me put it this way. Your construction, I think, in claim 54 is that you have to

mapped in the TNet. Couldn't do it. So there is no express teaching. You have to go to obviousness. And the obviousness argument was not made in their original petition. This is something they are now throwing out there. You won't find it in the original petition.

What we do know, and again, I have to argue this through attorney argument because we don't have the expert evidence and I apologize, Judge Tierney, but we know. This piece from ServerNet 2, which if you look at the front page of ServerNet 2, it says it's backwards compatible. This is as close as we get to a true understanding of how TNet works. You don't see that hourglass.

Remember I said the hourglass, you start with the PCI, you got the parallel bits, serialize them, get them to the other side, push them back out. What do we know? We know that the first time that the local physical address exists, the first time you know the physical address for that printer down over here is at the end of the process. And you can see it over on the left-hand side. Once at the destination N node the address is translated to produce the local physical address. Remember the Scrabble cubes, it's actually made up at the far end because what are you using? Address, validation and translation table that is in that logic on that bus.

So the first time you ever see a parallel PCI or parallel PCI address or any type of PCI address is on the bottom end. It's